

Programação Visual

Trabalho de Laboratório nº 5

Objetivo	MVC – Introdução. Aplicações básicas com acesso a base de dados utilizando a <i>Entity Framework</i> .
Programa	Pretende-se criar o site para a gestão de equipas nos Hospitais.
Regras	Implementar o código necessário e testar no fim de cada nível. Use as convenções de codificação adotadas para a linguagem C# e para o modelo MVC.
Nível 1	<ul style="list-style-type: none">• Crie um novo projeto ASP.NET Core Web Application (Model-View-Controller) com o nome HospitEST.• Defina numa pasta Models as seguintes classes e propriedades:<ul style="list-style-type: none">○ Hospital – Id, Name, Localization.○ Doctor – Id, HospitalId, Name, Practice, PracticeYears○ Patient – Id, DoctorId, Name, DateOfBirth, Pathology• Acrescente às classes criadas as seguintes propriedades de navegação:<ul style="list-style-type: none">○ Hospital – Doctors (List<Doctor>)○ Doctor – Patients (List<Patient>), Hospital○ Patient – Doctor• Para criar e visualizar os pacientes, crie um controlador com o nome PatientsController, usando <i>scaffolding</i> com <i>Views</i> e <i>Entity Framework</i>:<ul style="list-style-type: none">○ Nome: PatientsController○ Context: HospitEST.Data.HospitESTContext○ ModelClass: Patient• Adicione ao menu do site um link denominado “Pacientes” para a ação que lista os pacientes.• Agora que existe o modelo, é necessário criar a base de dados correspondente. Utilizando o Package Manager Console, execute os seguintes comandos:<ul style="list-style-type: none">○ Add-Migration HosDocPatMigration (após a execução deste comando, observe que uma pasta com o nome Migrations é criada contendo código C# para o MigrationBuilder)○ Update-Database (após a execução do segundo comando, a base de dados HospitEST é criado no SQL Server. No menu View do Visual Studio, seleccione o SQL Server Object Explorer e procure dentro de Databases pela base de dados criada)• Compile o projeto e verifique se existem problemas, e se a nova página está a funcionar sem criar pacientes.• Substitua o elemento <footer> da página _Layout por uma secção com o nome Footer utilizando o <i>html helper RenderSection</i>. Parametrize-a como não obrigatória (Required: false).• Para completar a estrutura inicial do site, remova do menu os links para a HomePage, Privacy e coloque na View Index uma “<i>@section Footer</i>” contendo o texto anteriormente substituído.

Programação Visual

Trabalho de Laboratório nº 5

Nível 2

- Crie os controladores para os médicos e para os hospitais como feito no nível anterior.
- Adicione à barra de menus os *links* para a lista de hospitais e de médicos à semelhança ao que fez para os pacientes.
- Teste **sem adicionar hospitais ou médicos** e verifique que ambas as páginas são abertas corretamente.
- Não se pretende utilizar o controlador de Hospitais para gerir os mesmos. Sendo assim, remova do controlador todas as ações, **exceto a ação Index e Details**. Remova também as vistas associadas às ações eliminadas. Na vista **Index** remova os links existentes para as mesmas ações.
- Antes de testar, aceda à base de dados usando a ferramenta **SQL Server Object Explorer** e adicione manualmente dois hospitais:
 - **Hospital da Luz (Lisboa)**
 - **Hospital do Outão (Setúbal)**

*Nota: Para aceder a esta janela clique na opção "View" no menu de navegação do Visual Studio e selecione o **SQL Server Object Explorer**.*

- Teste a aplicação.

Nível 3

- Utilize *data annotations* para restringir os dados inseridos nos campos dos modelos anteriormente criados. As restrições são as seguintes:
 - Para os **hospitais**: o **Name** e a **Localization** são obrigatórios e o nome não pode exceder os 20 caracteres.
 - Para os **médicos**: o **Name**, **Practice** e **PracticeYears** são obrigatórios e o **PracticeYears** não pode ser negativo.
 - Para os **pacientes**: o **Name** e **DateOfBirth** são obrigatórios e o **DateOfBirth** não pode ser do futuro.

Nota: para as datas, terão de utilizar [custom attributes](#)
- Teste a aplicação e verifique que as validações estão a funcionar.
- Os títulos no cabeçalho das tabelas não estão em português, corrija-os nas classes do modelo recorrendo à anotação **Display(Name="...")**.
- Na listagem dos pacientes altere os Ids dos médicos para o nome dos médicos e na listagem dos médicos faça o mesmo para o hospital.
- Na página de detalhes do hospital, altere o nome da página para ser o nome do hospital e liste também o nome dos médicos associados ao hospital.
- Nos detalhes do médico, faça também uma listagem de pacientes associados ao médico e faça com que ao clicar no nome do Hospital, redirecione para os detalhes do respetivo Hospital.

Programação Visual

Trabalho de Laboratório nº 5

Nível 4

- À semelhança do laboratório passado, iremos preencher a base de dados com informação de teste para uma fácil validação mais rápida do que foi desenvolvido. Dentro da classe que criou para o contexto, faça o *override* do método [OnModelCreating](#) e adicione o código necessário para inicializar a base de dados com **2 Hospitais, 4 Médicos e 8 Pacientes**.
- Para verificar que o método de inicialização está a funcionar corretamente, apague a base de dados através do *SQL Server Object Explorer*, apague a pasta com **Migrations** e, finalmente, volte a executar os comandos do nível 1 no **Package Manager Console**.
- Teste.

Nível 5

- Na view **Index** do controlador dos **Médicos** acrescente um link **Pacientes** a cada médico que leva o utilizador a uma nova view que **lista** os pacientes do médico selecionado.
- Na **View** que apresenta a lista de todos os pacientes adicione a possibilidade de ordenar os mesmos por médico ou por data de nascimento (ascendente e descendente).
- Adicione paginação à view de pacientes, esta deve trazer 5 pacientes por página.

Desafio

- Faça a listagem do primeiro ponto do Nível 5 reutilizando o método **Index** do controlador dos pacientes.

Notas

Para os identificadores siga as convenções adotadas pelo C#, nomeadamente:

- A notação camelCase para o nome das variáveis locais e identificadores privados.
- A notação PascalCase para os nomes públicos dos métodos e classes
- Não utilize o símbolo '_' nos identificadores
- Não use abreviaturas